

AIP HG 5200_

PEER-TO-PEER ELECTRONIC CASH REVISITED

BY CODE, FOR ALL

Andrew Kessler

May 2, 2025

Contents

| | | |
|----------|---|-----------|
| 1 | Introduction | 4 |
| 2 | Understanding currency requirements | 5 |
| 2.1 | Abstracting taxonomic assessment | 5 |
| 2.2 | Mathematical descriptors: Currency variable list | 6 |
| 2.3 | Dimensional analysis of price stability | 7 |
| 2.3.1 | Relevant Variables | 7 |
| 2.3.2 | Dimensional Relationships | 7 |
| 2.3.3 | Interpretation and Conclusion | 8 |
| 2.4 | Implementation requirements for electronic currencies | 8 |
| 3 | Economic growth | 9 |
| 4 | Currency supply management | 14 |
| 4.1 | Supply change functions | 16 |
| 4.1.1 | Block reward | 16 |
| 4.1.2 | Burn policy | 17 |
| 5 | Currency Information | 17 |
| 6 | Underpinning Digital Currencies via Computational Resource | 18 |
| 7 | Consensus Mechanism | 18 |
| 8 | Design choices | 19 |
| 8.1 | Handling division | 20 |
| 9 | Closing remarks | 21 |

1 Introduction

The first truly production-ready example of peer-to-peer electronic cash was, unmistakably, Bitcoin. It combined insights from Chaumian-style blind signing schemes to secure transactions against malicious actors and incorporated Backian-like pricing functions to align social incentives, protecting the network from manipulation, such as double-spending.

This technology stack proved that a payment network could reliably process transactions across an autonomous, decentralized system, even when some nodes were unreliable.

While Bitcoin has demonstrated that digital cash is possible (and remains a valuable asset, and technological success) it falls short as an ideal currency. This is not a question of inflation, adoption, or scalability but a fundamental issue of design:

- Bitcoin's fixed supply (21 million coins) incentivizes hoarding rather than spending.
- A good currency should maintain a stable value, but Bitcoin's price is notoriously volatile.
- Most businesses and individuals do not price goods and services in Bitcoin.

An ideal currency should serve as:

1. A **medium of exchange**
2. A **store of value**
3. A **unit of account**

By this definition, Bitcoin does not function as a fiat substitute. In fact, most cryptocurrencies do not attempt to behave like fiat; they act more like digital stocks or equities, deliberately designed for volatility. Even stablecoins (designed to serve as a unit of account, store of value, and medium of exchange) fail to function autonomously.

This proposal seeks to address these shortcomings by introducing a new approach: programmable, autonomous, unpegged, non-collateralized, and non-game-theoretic digital currency with price stability within an order of magnitude, offering a true alternative to fiat. Only then can we transition from *"In God we trust"* to *"By code, for all."*

To achieve this, we need a **consensus-driven computational system** capable of processing complex analytics. This is where **generative AI** within the consensus mechanism comes into play. By combining **blockchain-based rule sets** with **AI-driven analytics**, we unlock the possibility of programmable, autonomous fiat alternatives.

Finally, we apply sound currency principles (rooted in millennia of economic wisdom) executed through **new, innovative strategies**.

2 Understanding currency requirements

Paul Samuelson once remarked, "*When a technology becomes sufficiently advanced and ubiquitous, it disappears.*"

This phenomenon is often referred to as *the invisibility of mature technology* or *technological normalization*. Examples include fire, the wheel, electricity, and even the internet or smartphones, tools that have become so deeply integrated into daily life that they no longer feel like "technology."

A more formal framework for this concept is sometimes called *the Clarke-Samuelson Technology Pyramid* or *the technology adoption ladder*. The idea is that once a technology is fully embedded in society and taken for granted, it ceases to be recognized as a distinct technological innovation.

Cash is a prime example. Having evolved in form and function over millennia, cash is often analyzed through a rigid taxonomic lens, meaning we study existing currencies like the dollar or the yen by asking, *What do they have in common?* However, this approach has a critical flaw: **inheritance bias**.

Inheritance bias means that the shared traits of successful currencies merely reflect the status quo, not necessarily the optimal design for money. By focusing on what has worked in the past, we risk missing opportunities to develop superior monetary systems, especially in the digital era. The challenge is not simply to refine what already exists but to rethink currency from first principles—without being constrained by legacy assumptions.

This is a complex problem, as it requires identifying *what we don't yet know*. To navigate this challenge, we can apply three key approaches:

- **Abstracting a taxonomy** – Using surface taxonomy to identify specific behaviours or properties is not useful but for abstracting concepts it is a good anchor (e.g., generalizing a category like "interest rates" into broader concepts like "borrowing and lending").
- **Mathematical description** – Identifying the key variables that influence currency behavior and price stability as a clean, non-linguistically description of factors that matter.
- **Dimensional analysis** – Dimensional analysis is a mathematical technique used to estimate relationships between variables when the exact equation is unknown. We use this as a final check. We pretend the relationships are not known and use dimensional analysis to keep us honest and to focus on the quantities that have the most impact on currency design.

2.1 Abstracting taxonomic assessment

A taxonomic approach to currency design reveals that successful currencies tend to exhibit the following properties:

- Trust and Legitimacy

- Social and Political Stability
- Adaptability and Governance
- Supply and Demand Dynamics
- Exchange Mechanisms
- External Exchangeability
- Production and Value Creation
- Borrowing and Lending
- Consensus on Unit of Account
- Energy and Resource Backing

From these, we can generalize and abstract key principles:

- Information fidelity, veracity, and accessibility
- Monetary momentum (supply and velocity)
- Medium of exchange efficiency
- Mechanisms for financing growth
- Price signaling capacity
- Economic underpinnings (backing by resources or energy)

Ultimately, what emerges is a broader, more fundamental insight: *A good currency is one that effectively supports healthy market functions.* This provides a qualitative framework for assessing monetary systems.

2.2 Mathematical descriptors: Currency variable list

Now, shifting from a qualitative assessment to a quantitative one, we can explore the mathematical properties of money. By reviewing monetary functions such as the Fisher equation, inflation rates, the Taylor rule, and the quantity equation of money, we can derive a structured list of key variables that influence monetary stability and price dynamics:

- (M) Money Supply
- (V) Velocity of Money
- (P) Price Level
- (Y) Real Output
- (π) Inflation Rate
- (i and r) Nominal and Real Interest Rates
- (π^e) Expectations of Inflation
- Output and Inflation Gaps

2.3 Dimensional analysis of price stability

Price stability of the dollar is influenced by various economic factors, including inflation, money supply, and economic output. Using dimensional analysis, we derive a fundamental equation that governs price stability.

2.3.1 Relevant Variables

We consider the following key variables:

- **Money Supply** (M): Total supply of money in circulation. **Dimension:** $[M]$.
- **Velocity of Money** (V): The rate at which money circulates in the economy. **Dimension:** $[T^{-1}]$.
- **Gross Domestic Product** (GDP, Y): Total economic output measured in dollars per unit time. **Dimension:** $[MT^{-1}]$.
- **Price Level** (P): General price level of goods and services. **Dimension:** $[M^{-1}]$.
- **Inflation Rate** (π): The rate of change of prices over time. **Dimension:** $[T^{-1}]$.

2.3.2 Dimensional Relationships

A fundamental equation in monetary economics is the **Quantity Theory of Money**:

$$MV = PY. \quad (1)$$

Since price stability depends on inflation, we assume:

$$\pi \sim f\left(\frac{dM}{dt}, \frac{dY}{dt}, M, Y\right). \quad (2)$$

The dimensional analysis gives:

$$\begin{aligned} [M] &= M, \\ [Y] &= MT^{-1}, \\ \left[\frac{dM}{dt}\right] &= MT^{-1}, \\ \left[\frac{dY}{dt}\right] &= MT^{-2}, \\ [\pi] &= T^{-1}. \end{aligned}$$

To construct a dimensionally consistent equation:

$$\pi \sim \frac{1}{P} \left(\frac{dM/dt}{Y} - \frac{dY/dt}{Y} \right). \quad (3)$$

Thus, introducing a proportionality constant k :

$$\pi = k \left(\frac{dM/dt}{Y} - \frac{dY/dt}{Y} \right). \quad (4)$$

2.3.3 Interpretation and Conclusion

The equation suggests:

- $\frac{dM}{dt}$ represents the rate of change of money supply.
- $\frac{dY}{dt}$ represents the growth rate of the economy.
- If $\frac{dM}{dt} > \frac{dY}{dt}$, inflation increases ($\pi > 0$), causing price instability.
- If $\frac{dM}{dt} = \frac{dY}{dt}$, price stability ($\pi = 0$) is maintained.

Thus, for a useful currency, the money supply should grow at the same rate as real economic output:

$$\frac{dM}{dt} = \frac{dY}{dt}. \quad (5)$$

2.4 Implementation requirements for electronic currencies

From our analysis and consideration above we can generate a list of properties and behaviors that are most important towards building an electronic replacement for fiat.

- Efficient and stable economic growth.
- Effective dynamic supply management.
- Veracity, fidelity and easy access to monetary information.
- Underpinned by computational resource
- Choice of consensus mechanism.

Economic growth in turn requires robust financing with an appropriate machine-level understanding of risk, the offer of goods and services in exchange for the electronic currencies, and marketplaces and tools where one can search for, find and build products.

Effective dynamic supply management requires an autonomous method for establishing monetary policy under consensus and a mechanism to increase and decrease currency supply.

Access to monetary information means that all economic and financial systems built on top of the smart market and smart cash frameworks can compute the currency price, market depth, price slippage, etc can all be calculated procedurally. The network does not have to trust that the data they consume is reliable. They can compute it themselves.

Underpinned by computational resource. Coming from the traditional world, we see that an autonomous programmable fiat replacement can be backed by instruments, real world assets, digital goods and services, but in a pure and abstract sense, it is meaningful to note that what really underpins the network is energy and compute power. The GPU network that runs the AI analytic models, hashes blocks and the relay and storage nodes that provide communication and data, these are the fundamental value backers of the network and the currency. We might imagine that the US dollar is (to some extent) backed by faith and oil,

then electronic currencies should be backed by provable information (not faith) and compute resource (not oil).

Choice of consensus mechanism allows electronic currency designers to build their economic systems based on capital or participation. We have elected the participation model, the ability to contribute useful work, not once off capital, is our preferred basis. Again, as seen in the dimensional analysis, economic growth, not capital, is the primary driver for establishing fiat replacement currencies. Capital is critical, make no mistake, what we mean is that capital already exists and can be ported easily into the system. But because the choice of consensus mechanism is mutually exclusive, rewards for maintaining network security, providing services, etc should be based on continual effort for continual reward. This prevents monopolies and collusion from emerging in an undeserved manner. Thus we have adopted Proof of Work and modified it to be more energy efficient, rather than electing Proof of Stake.

3 Economic growth

Our design for a fiat replacement cryptocurrency is tightly tied to economic growth. This is the seat of value within the system. By understanding how the network supports **INVEST**, **CREATE** and **SELL** functions within smart markets and that these signals and data is what drives the currency supply, we better understand why this electronic currency is a scalable and suitable alternative to fiat.

It is worth noting that we do research and consider models for economies that experience net-zero growth rates and even stagflation. These are important and difficult problems to solve for and may in many cases be impossible to support, but to compete with fiat backed economies on a like-for-like basis, it is the condition of growth for which we want to excel at.

Investing in a digital economy comes in two forms. Finance and Digital infrastructure. First we describe our financial instruments framework, then our GPU, storage and compute investment into the economic growth of the system.

ACTUS as a new Electronic Financial Standard

The industry currently uses instruments like ERC-20 or BEP-20 for financing. DeFi contracts are also popular. However, these instruments are not built on an understanding of risk, debt, earnings at risk or value at risk. Traditional financial institutions cannot accept such instruments because, not because they lack familiarity with the choice of standard but poorly instruments can have unintended behaviors that affect price, not because of a real price shift, but because of the instrument itself. These standards do not have an understanding of "good debt" versus "bad debt". Certainly the rate of inflation has nothing to do with these instruments and they do not provide critical "health check" data on the economy they are integrated into. A lack of careful design and standardisation of financial instruments can lead directly to recession and market collapse as we saw in the 2007–2010 subprime mortgage crisis. Even in traditional finance there is significant need to automate reporting and compliance on things like the Dodd-Frank act, provide methodologies and consistency across finance. Thus we conform to the ACTUS framework to implement system wide finance for better, more efficient, more scalable economic growth which in turn strengthens the currency.

ACTUS (Algorithmic Contract Types Unified Standards) provides a unified, algorithmic way to represent nearly all financial contracts, bringing consistency and transparency to electronic transactions. By encoding contracts in a standard data format plus algorithm, ACTUS can automatically generate the precise schedule of cash flows and obligations over a contract's life. This precision reduces ambiguity and errors, making it easier for institutions to process transactions and for all parties to understand their commitments. In essence, ACTUS lets **finance speak one language** by ensuring that a loan, lease, or derivative means the same thing everywhere and produces expected outcomes with mathematical certainty.

Comparison to SWIFT, ISO 20022, and FIRD

Traditional standards like SWIFT messaging and ISO 20022 focus on ***communication*** – they standardize how information is exchanged between institutions, but not the full economic behavior of a contract. For example, ISO 20022 is a global messaging standard that provides consistent, structured data for financial transactions (e.g., payments, securities trades). This improves data quality and reduces operational risk in message processing, but it doesn't define how the cash flows of a loan or bond are calculated.

By contrast, ACTUS standardizes the ***content and computation*** of financial contracts themselves – it models “who owes what to whom, and when” for every contract with rigorous algorithms. This means risk events (like interest resets, amortizations, defaults) are handled in a uniform way by the ACTUS algorithms, rather than being described in free-form text or bespoke code. The OFR's Financial Instrument Reference Database (FIRD) highlights this complementarity: it adopted ISO 20022 as one reference for data fields and then added ACTUS as a second reference to standardize contract definitions across equities, debt, derivatives, etc.

In short, ISO 20022/SWIFT cover the messaging **syntax**, whereas ACTUS defines the contract **semantics** and cash-flow logic – together improving efficiency and reducing risk. ACTUS's uniform contract templates let institutions automate processing beyond just messaging, achieving consistency from transaction initiation to settlement and reporting, something pure messaging standards cannot fully deliver.

Supporting a Price-Stable Cryptocurrency

A cryptocurrency aiming for stable value (for example, one backed by commodities or other valuable assets) stands to benefit from ACTUS as its financial contract standard. Such a crypto could be designed so that each coin is linked to underlying financial contracts – e.g., loans for commodity production, asset leases, or other cash-flow generating agreements. ACTUS provides a ***structured framework*** to represent and automate these contracts, ensuring that the coin's backing assets are transparent and their cash flows predictable.

This means the stablecoin's reserve contracts (say a loan for gold mining or an oil inventory financing deal) would all be encoded with the same rigor: interest rates, repayment schedules, and triggers are clearly defined in ACTUS terms. Investors and regulators could then easily verify the health of the stablecoin's backing: the standard algorithms would project future cash inflows from the assets, and any shortfall or risk can be spotted early. In broad terms, ACTUS enables ***tokenization of real assets*** by translating traditional contracts into machine-readable, executable form. This supports a crypto ecosystem where digital tokens are directly tied to real economic activities.

The benefit is twofold: (1) ***Stability*** – the crypto's value is anchored by real asset cash

flows that are contractually enforced and transparent; and (2) **Trust and compliance** – because ACTUS is precise and open, audits or oversight of the stablecoin’s asset contracts are straightforward, which can improve accountability and reduce the likelihood of hidden risks. Compared to ad-hoc or opaque asset backing, an ACTUS-based stablecoin would operate more like a well-regulated fund – every coin is a claim on a set of standardized contracts, making its value far less prone to sudden loss of confidence.

Mitigating Financial Crises with ACTUS

A major promise of ACTUS is improved risk monitoring and crisis prevention through better data. Financial crises often spread because no one has a clear, timely view of complex contractual exposures. For instance, during the 2007–2010 subprime mortgage crisis, loans were repackaged into complex derivatives (CDOs, MBSs) in non-transparent ways. When defaults rose, **uncertainty** about “who is exposed to what” froze credit markets.

ACTUS’s structured framework would have drastically improved transparency in such a scenario. If all those mortgages and CDOs had been encoded in ACTUS, each bank or regulator could **algorithmically aggregate** the cash-flow obligations and see exactly which contracts (and institutions) would suffer under, say, a 20% drop in home prices. By dividing contracts into standardized types based on their cash-flow patterns, ACTUS makes it easier for algorithms to “read” and aggregate obligations across millions of contracts.

Regulators can conduct stress tests in days or hours instead of weeks, because the data needed is already standardized and machine-ready. This early warning capability means interventions (like requiring more capital or unwinding risky positions) can happen **before** a crisis spirals. Moreover, ACTUS enables **systemic risk assessment** by directly quantifying how institutions are interconnected via contracts. In an ACTUS-enabled ecosystem, a contract in default would be instantly recognized in all counterparty risk calculations, preventing the kind of blind contagion seen in 2008.

Overall, using ACTUS in electronic financial instruments builds a more resilient financial system: enhanced transparency, precise risk data, and automated monitoring act like a financial immune system, spotting vulnerabilities and enabling informed action to avoid widespread failures.

Investing computational resources

In a world driven by electronic currencies and digital economies, the question arises: How do AI agents, developers, and digital asset creators (pharmaceutical drug discovery, game developers, animators, musicians etc) get financed? Traditional financial models struggle to integrate compute as an investable asset. However, a robust and distributed GPU network of miners running a PoW based consensus mechanism offers a unique opportunity. These miners can not only secure the blockchain network, but also provide computational resources for compute services, similar to how lenders invest capital. We explore here the concept of a Compute-Backed Financial Instrument (CBFI) that allows miners to stake compute, earn returns, and support compute-driven economies.

Investment Model: Compute as an Asset Class

Similar to traditional finance, where capital is invested in businesses, GPU compute power can be “staked” into a **Compute Investment Pool (CIP)**. This enables developers, AI agents, and digital service providers to access high-quality compute resources through structured financial agreements.

Key Components:

- **Mining Nodes as Lenders:** GPU miners stake compute power into a smart contract, which is tokenized as a CBFI asset.
- **Developers as Borrowers:** AI and digital service developers utilize this staked compute to power applications.
- **Decentralized Governance:** DAOs or smart contracts distribute compute resources based on demand and risk assessment.

Repayment: Monetizing AI and Compute

To ensure miner-investors receive returns, various monetization models are established:

Revenue Models for Compute Investment:

- **Subscription-based AI Services:** Compute-backed projects charge users for AI-driven solutions and distribute earnings to miners.
- **Pay-per-query API Models:** Decentralized AI services monetize query-based compute usage.
- **Tokenized Compute Yield:** Compute stakers earn a percentage of transaction fees from AI services.
- **Compute-backed Stablecoins:** Compute resources collateralize a digital currency used within AI ecosystems.

Yield Distribution to Miner-Investors:

- Revenue from AI agents is automatically shared with GPU miners.
- Miners may acquire fractional ownership in AI startups.
- Time-locked bonds redeemable for future yield are issued based on staked compute.

Economic Growth and Scaling

This model unlocks significant economic potential by decentralizing access to compute resources:

Economic Benefits:

- **AI Innovation:** Startups and digital creators can scale AI applications without large capital expenditures.
- **Product generation:** As with financing agricultural commodities that take both capital and seed as input, CBFI take capital and compute as input but produce content, data, products that are then sold into the market creating more demand for the underlying currency.
- **Job Creation:** AI-powered digital economies create new employment opportunities in AI development and content creation.

Some economists have proposed energy-backed monetary systems, where a currency is tied to joules, kilowatt-hours, or barrels of oil. The nuanced impact of this insight is that we forgo the evolutionary step of energy backing and go straight to clock cycles or compute backing. As oil indirectly backs the dollar, silicon and compute efficiency indirectly backs a currency where such CBFI contracts flourish. The author notes that with the advent of RISC GPU architectures poised to impact medical diagnostics, sensor arrays, autonomous vehicles and personal robotics, CBFI will dramatically impact and strengthen associated currencies.

Risk Management

To ensure sustainability, risk must be effectively managed:

Key Risks and Mitigations:

| Risk | Mitigation |
|-------------------------|--|
| Compute Volatility | Dynamic pricing via smart contracts |
| AI Project Failure | Diversified compute staking across multiple projects |
| Miner Liquidity Lock-up | Secondary markets for compute-backed tokens |
| Centralization Risk | DAO governance for fair compute distribution |
| Fraud and Default | Smart contract-based escrow and collateralized compute |

Conclusion

The Compute-Backed Financial Instrument (CBFI) represents a new financial primitive that bridges DeFi and AI or compute infrastructure. By allowing miners to invest in compute-driven services and digital economies, this model:

- Enables miners to generate yield beyond traditional token emissions.
- Democratizes access to high-performance compute resources.
- Creates a compute-backed digital currency for AI and compute-powered economies.

This framework paves the way for a future where decentralized AI and computational resources form the backbone of a self-sustaining digital economy.

Creating machine readable assets

Machine readable assets (MRAs) are a new set of assets that exist as part of a file system and can be governed by any public-private key system. There are many offerings that provide equivalent functionality but are not considered MRAs. For example IPFS, Storj or smart data projects all provide some degree of amalgamation between data management and digital rights management, but MRAs are different. MRAs define the owner of some asset separately from the content creator and are in principle deployable over any blockchain style system with no modifications required. MRAs are any binary object that is distinct from and linked to a public-private keypair. A machine-based process can generate a public key that a user can use to assign data management and digital rights management to some binary thing. This means that real world assets, game skins, contracts, identity can be made machine readable in a consistent and scalable way. This is how developed goods and services are represented in the system and signal the systems capital.

While ACTUS and CBFI represent economic growth, MRAs represent capital. Thus they can provide collateral and have functional equivalence to property.

Marketplace node

The Consumer Price Index (CPI) is a measure that tracks the average change in prices paid by consumers for a basket of goods and services over time. It is a key indicator of inflation and purchasing power within an economy. It is one example of an index that is crucial in establishing things like policies on inflation or interest rates. However it is a very hard task to establish the validity of a CPI and the information is seldom available in real-time, often it is significantly retrospective.

Thus, the presence of a marketplace node within a layer one infrastructure is not just a convenience in the retail experience but provides data and insight into price formation, clearance and settlement, CPI and more.

Marketplace nodes allow the system to track performance of ACTUS finance, CBFIs, MRAs to assist the miners to uniformly adjust their monetary policy functions such as currency supply.

4 Currency supply management

Currency supply requires sound analytical frameworks acting on trustless information to compute fiscal policy for issuance. This computation needs to be performable by anyone using the network and for all participants to agree on the output. When Satoshi Nakamoto released Bitcoin in 2008 this was not technically feasible and so the issuance function simply returned a fixed number (21 million) as the currency supply target. With the advent of Gen AI (automating and reducing the cost of intelligence) and vector auto-regression (VAR) based fiscal policy frameworks, fiscal policy can now be held under consensus in a feasible computable manner.

Currency supply management requires miners to first agree the policy through self computing it from a common agreed upon data set, and then enforce one of two methods.

- Method 1: - Increase the supply target which in turn amends the block reward.
- Method 2: - Decrease the supply by imposing a transaction fee fraction that is payable as an input into a transaction with no corresponding output effecting a fractional burn which over time gradually reduces the circulating supply.

Critically, none of these functions are issued from a DAO or central control body, but executed in a fully distributed way autonomously and consistently. Thus currency supply management is achieved through miner based distributed analysis using FAVAR, a supply increase function and a supply decrease function.

Factor Augmented Vector Auto Regression or FAVAR, as developed by Ben Bonanke and the National Bureau of Economic Research NBER is a suitably sophisticated fiscal policy model, executable on GPU based architecture and approximates well over machine learning approaches.

Here we review some important aspects of FAVAR and review how it is implemented for fiat currencies that are released into circulation from central banks. Once we understand how FAVAR is implemented, we will discuss how the function of central banks is subsumed by special node types within the network, combined with a DEX listing to provide "foreign currency pairs" and how the aggregate price of these nodes is the AIBX coin price. Lastly, we

take all of this critical data as inputs into a GEN AI agent that runs the FAVAR model to issue the supply policy recommendations.

The FAVAR framework essentially extends the traditional VAR approach by “augmenting” it with latent factors that are estimated from a large panel of macroeconomic time series. In the context of designing a monetary policy to stabilize a currency’s price, FAVAR is applied as follows:

Rich Information Capture.

Rather than relying on a few key variables (as in standard VARs), FAVAR uses many observable series (for example, measures of output, inflation, employment, financial market indicators, and—in this case—exchange rates) to extract common factors. These factors can be interpreted as capturing underlying economic forces such as overall economic activity, price pressures, or liquidity conditions that are relevant to both domestic and international financial markets. This richer information set allows policymakers to better understand how shocks (for instance, a change in the policy interest rate) affect the economy—including the currency’s price—by filtering out noise and measurement error present in any single series.

Policy Impact Assessment via Impulse Responses.

Once the model is estimated, one can compute impulse response functions to trace the dynamic effects of a monetary policy shock on the exchange rate (among other variables). By analyzing these responses, a central bank could simulate different policy adjustments and gauge which shocks or interventions (such as a 25-basis-point change in the policy rate) lead to stabilization of the currency’s value. Essentially, the model helps in “calculating” the expected outcome of a policy change on the exchange rate over time.

Data Requirements.

To implement FAVAR for currency stabilization, one needs:

- A key monetary policy indicator (e.g., the domestic policy rate)
- A broad set of “informational” time series that include not only domestic economic indicators (like GDP, industrial production, inflation, money aggregates) but also financial variables that may directly or indirectly affect the exchange rate (such as asset prices, credit spreads, or even international variables if available).
- If the target is the currency’s price (or exchange rate), then it’s crucial that the data set either includes direct measures of the exchange rate or contains indicators closely related to currency valuation dynamics.

Feasibility and Practical Considerations.

While the FAVAR approach is more computationally demanding than standard VARs (especially if one uses likelihood-based methods like Gibbs sampling), it is feasible with modern computing power. The method’s strength lies in its ability to harness a wealth of information, thus the feasibility also depends on having high-quality, high-frequency data that accurately reflect the economic and financial conditions influencing the currency. Because of the nature of ACTUS described above and that the network has access to marketplace nodes, this information is far more available and reliable than would be present in fiat based, central bank backed currency models. In addition, the identification of monetary policy shocks (i.e., separating genuine policy moves from responses to incoming economic information)

remains a challenging but critical task which can uniquely be adjusted for through training in the ML analytics pipeline.

In summary, FAVAR provides a promising framework for designing monetary policies aimed at stabilizing a currency's price because it integrates extensive economic information into the analysis, allowing policymakers to simulate and assess the dynamic responses of the exchange rate (and other variables) to various policy interventions. Its success depends on both the quality and breadth of available data and the careful identification of structural shocks. This is more difficult to do in practice with fiat but far more practical in electronic network based currencies, linked through transparent usage, trade and financing data.

4.1 Supply change functions

The "current supply" of this currency can be calculated as is:

$$(\text{Current supply})_{n+1} = \sum_{\text{block}=0}^{\text{block}=n} (\text{block reward}_n - \text{burned coins}_n)$$

Where block reward and burned coins are a function of the difference between the supply policy target as calculated by FAVAR described above and the current supply.

$$\text{block reward}_n = f(\text{target}_n - \text{supply}_{n-1})$$

$$\text{burned coins}_n = f(\text{target}_n - \text{supply}_{n-1})$$

Thus, a change in the currency supply can be changed simply by adjusting the target to the FAVAR computed recommended target.

4.1.1 Block reward

$$\text{Block reward} = \text{frac} * \begin{cases} (((\text{Target} - \text{current supply}) \gg m) + k) & \text{curr sup} < \text{target} \\ k & \text{curr sup} \geq \text{target} \end{cases}$$

The block reward function has two behaviors. One for when current supply < target and another such that supply ≥ target. *curr sup* is understood as shorthand for "current supply". *k* and *m* are quasi constants. while in principle these values can be adapted they determine the "half life" of any potential policy shocks and are logarithmic to the target. » denotes the bit-wise operation right shift (RSHIFT) common in Bitcoin and Monero style issuance functions. "frac" is the number of fractions per coin. For clean presentation the fraction multiplication is performed on the bracket but in computation the multiplication must be done before the RSHIFT operation. One must bear in mind that because of the recursive nature of this function certain *k* and *m* values might trigger random behavior as is the case with all "logistic map" problems. Should this become problematic we will add an Exponential Moving Average (EMA) method such as a low-pass filter like:

$$\hat{X}_i = \alpha X_{i-1} + (1 - \alpha) \hat{X}_{i-1}$$

in order to reduce jitter or prevent random behavior.

4.1.2 Burn policy

Defending a percent or proportional fractional burn policy set as a transaction fee is easy to defend. In game-theoretic terms, proportional burns are **strategy-proof** with respect to transaction splitting: paying in two 50-unit transactions vs. one 100-unit transaction yields the same total burn cost if the rate is flat 1. Thus, users have less reason to deviate from their natural spending patterns just to game the fee. From a macro perspective, a modest proportional burn can act like a continuous sink that counterbalances new supply or inflation, potentially contributing to long-term stability if tuned correctly. Whereas flat burn fees or tiered schemes (progressive or regressive) introduce incentives to restructure transactions. Favoring small transactions can promote inclusivity and discourage wealth concentration, but it risks splitting exploits. Favoring large transactions may improve efficiency for high-value uses, but it unfairly advantages the wealthy and could encourage centralization of wealth.

$$\text{Burn tx fee} = k \mid \text{TX value} \mid$$

Thus the burn policy is simply a percentage of the magnitude of the payment.

There must be a maximum roof to k to ensure that one does not prohibit the normal spending of coins. This means that there are market conditions where a strict reduction in supply cannot be guaranteed. One can guarantee the reduction in the inflation rate and one can encourage reduction assuming that there is enough transaction volume such that the sum of the burned coins exceeds the block reward.

5 Currency Information

In traditional fiat systems, currency policy models are based on imperfect and often siloed information, yet they function to some extent. However, our system benefits from a significantly improved approach due to structured financial contracts (ACTUS), real-time marketplace data, and machine-readable assets.

By utilizing ACTUS-structured financial contracts, we enable precise economic growth analysis. Moreover, marketplace nodes provide real-time information on price formation, trading volumes, and liquidity. This machine-readable asset framework allows the system to extract high-quality data and conduct rich analytics on the state of the economy.

This transparency is critical for adjusting monetary policy dynamically and strengthening currency stability. Unlike fiat systems where data manipulation and lack of transparency often hinder accurate policy adjustments, our approach, leveraging Factor-Augmented Vector AutoRegression (FAVAR), ensures that currency stabilization efforts are data-driven and efficient. The ability to self-compute economic health from open, verifiable sources removes reliance on opaque central authority data models, making our system fundamentally more resilient and adaptive compared to traditional fiat structures.

6 Underpinning Digital Currencies via Computational Resource

A unique and fundamental aspect of our system is that the same underlying computational resource used for network consensus also underpins the creation and validation of financial instruments. This synergy ensures a tightly integrated economic model where currency supply and economic growth are inherently linked.

Miners providing computational resources do more than just validate transactions; they also:

- Generate rich ACTUS financial contracts, which define structured economic agreements.
- Validate and maintain machine-readable assets (MRAs), which represent real-world and digital economic value.
- Support Compute-Backed Financial Instruments (CBFI), enabling compute-based lending and financial structuring.
- Provide the economic foundation for currency pricing by ensuring a robust and decentralized computational economy.

This results in a singular system that organically generates both sides of a market in equilibrium. On one hand, computational resources validate and structure financial instruments, ensuring economic stability (the "value" arm). On the other, the same resources underpin currency supply mechanisms (the "pricing" arm). This intrinsic linkage is highly unique in digital currency design—traditional financial systems do not derive their value from an underlying computational mechanism, nor do they integrate economic growth with computational resource allocation in this way.

While it is the structured instruments, financial standards, and analytic methodologies that directly support currency value, computation itself acts as the fundamental backing mechanism. This approach departs from conventional financial models, where money is backed by trust or collateralized assets, and instead aligns digital currency with provable, quantifiable compute power.

7 Consensus Mechanism

Given our economic objectives, we have opted for a participation-based consensus mechanism—specifically, Proof of Work (PoW)—over a capital-based model like Proof of Stake (PoS).

PoW aligns incentives such that those who contribute useful computational resources are rewarded, ensuring a meritocratic network where security and economic utility are directly linked. Unlike PoS, where wealth accumulation disproportionately determines consensus influence, our approach values continual effort, preventing monopolization and reducing economic centralization.

Furthermore, GPU-based PoW uniquely provides the computational resources necessary to support ACTUS financial contracts, machine-readable assets, and Compute-Backed Financial Instruments. These components rely on distributed compute to remain verifiable, efficient, and decentralized.

Of course, a holistic perspective on energy consumption and scalability is necessary. Recognizing the concerns surrounding PoW's environmental impact, we have developed a novel refinement of the model that optimizes energy efficiency while maintaining security and decentralization. The details of this optimized PoW mechanism are discussed elsewhere, but as a fundamental design choice, our system prioritizes PoW due to its alignment with economic stability and computational resource allocation.

8 Design choices

Mathematics is fundamentally about establishing absolute truth through proofs. A proof is a logically rigorous argument that shows a statement is always true, independent of external conditions.

With the advent of practical and scalable AI methodologies. A lot of traditional paradigms do not have to apply. We need to bear these things in mind that while we revolutionize the basis for currencies we ensure that the approach is "future proof". To this end it is important to note the following design choices.

Computer science, particularly programming, focuses on processes—how things are computed, transformed, or solved step by step. It is about designing and implementing procedures (algorithms) that take input and produce output efficiently.

Mathematics proves that an algorithm works. Computer science implements and optimizes that algorithm in practice.

Within the AI revolution there is no need to distinguish between the functions, data and definitions. A prompt within a Gen AI approach will not only establish a kind of proof, but also find the relevant data within its learning and compute the answer.

With this in mind it might be possible with other usecases at the intersection between blockchain and AI that only a blockchain prompt is needed, such as "Based on your current training and the data of the system, compute an efficient monetary policy for the next six months and prove the efficiency of such an approach".

However, currency supply is a function of currency supply. This inherently recursive function can easily generate algorithmic entropy. algorithmic entropy refers to the complexity inherent in the algorithm itself rather than the randomness or disorder present in the data it processes. Unlike thermodynamic or Shannon entropy, which quantify disorder or information content in a dataset, algorithmic entropy arises from the structure and rules encoded within an algorithm. Even if the input data is simple, a complex algorithm can introduce entropy due to its own internal operations, decision processes, and state transitions. As a result, we must carefully constrain the "how" or process of supply increases or decreases.

Thus we predefine methods that implement the policies for currency supply management using a traditional comsci approach but we let the ML generate the monetary policy to remove imperative declarations providing an attack surface for human manipulation and strictly remove politics, sentiment or belief from such policy sets. Thus "In God we trust" may become "By Code, For All".

With such an understanding we specify the following design choices.

8.1 Handling division

Division within a set of integers \mathbb{Z} presents several challenges. When thinking about trade, payment for goods and services, currency pairs and more, the limits of currencies as integers are felt. Non-Closure: Unlike addition, subtraction, and multiplication, division is not closed over integers. For example

$$7/3 = 2.333...$$

which is not an integer. Thus three people cannot share 7 dollars.

This observation introduces the next problem to manage. Approximation and Rounding (for example with Floating Exchange Rates). Currency exchange rates fluctuate and are often rounded for practical use, similar to how integer division often rounds or truncates non-integer results. Rounding errors in finance can lead to discrepancies in accounting and pricing.

Thus we introduce two coin types. The main coin type which forms the basis of the AIBlock based economy is setup such that each coin has 72072000 fractions.

The choice of 72072000 is because 720720 is a "highly composite" number. A highly composite number (HCN), sometimes called an anti-prime, is a positive integer that has more divisors than any smaller positive integer. In other words, an HCN n satisfies the condition that no integer $m < n$ has more divisors than n . For example, 10 can only be divided by 1,2,5,10. 12, on the other hand can be divided by 1,2,3,4,6,12 making it a much better base for a currency. 720720 has 240 divisors, however it can only be divided by 10, not 100 or 1000 etc. Thus we compute a multiple of 720720 to correct this issue. 72072000 can be divided perfectly to 3 decimal points and has 240 divisors thus you can make a 0,001 AIBX coin payment just as easily as $1 / 3$ or a coin payment. Because the upper bound of a UINT 64 is 18446744073709551615 with 72072000 we have enough space in the register to be able to issue up to 255 billion coins. if we chose to support 4 decimal places we would run out of issuance at 25 billion which might genuinely be a problem within the first 10 years of operating a market coin. The reason being that market coins often times need to record massive cash flows even if the profit to market participants is far less. For example, a mere 2000 grain finance contracts in Brazil require 200 Million USD in cash flows. If we take an annualized return of 20 percent then the net gain is 40 million USD. The point is clear, 2000 small to mid sized contracts where the AIBX coin price is 20 USD cents could need 1 billion coins in circulation thus realistically we would need to consider the register overflow of a UNIT 64. While we can migrate to a U128 register if needed this is not without additional consideration.

With a base coin well designed we do need to consider a scenario where the coin price increases to 10 000 dollars. Here a 3 decimal place limit creates a challenge making the smallest possible payment to be 10 USD. In a micro-payments framework this is untenable. This is why we have introduced a second coin type, namely the Lepton.

As with our example above, if we had 7 AIBX coins we would have 504504000 fractions. This divides easily and perfectly by three to result in exactly 168168000 fractions. Thus sharing 7 AIBX amongst 3 people leave no remainder and no need to round or approximate.

They can divide perfectly in 3, 5, 7 pieces and more. They also accommodate 4 decimal points for FOREX and exchange rate purposes. This minimizes as far as possible rounding errors in the use of AB coins. The total number of fractions will be reported here in product notation (as in $\chi * 720720000$) to keep a strong intuition for the numbers. *Frac* is understood

as shorthand for "fraction number". All operations are performed on the number of integer fractions for precision but when displayed at wallet level the amounts are displayed as whole coins, ie x / frac . One can ignore the $\text{frac} * \text{terms}$ in the general equations for ease of understanding.

Each block is produced in exactly 30 second intervals giving 2880 blocks per day. The absence of a transaction fee market or staking contracts give us no economic activity or circulating supply oracle other than transaction volumes. Hence our burn policy relies only on this metric.

9 Closing remarks

Traditional cryptocurrencies like Bitcoin have struggled to function as true currencies due to fixed supply constraints, speculative volatility, and lack of real-world economic integration. The AIBlock research proposes a new kind of electronic cash: an AI-driven, dynamically managed coin with relative price stability that is unpegged, non-collateralized, and free from central bank manipulation. This currency, backed by a real, on-chain GDP of goods and services, can autonomously adjust its supply through AI-powered monetary policy. This ensures stability while maintaining decentralized issuance, removing reliance on centralized banking institutions and the traditional financial system.

This cryptocurrency would function as both a medium of exchange and a store of value, allowing seamless, low-cost transactions while maintaining price stability through Factor Augmented Vector Auto Regression (FAVAR) models. By dynamically adjusting supply based on real-time economic activity, this currency eliminates the need for speculative hoarding, thereby achieving true economic efficiency.

The implications are profound. By introducing an AI-backed, Smart Market-driven cryptocurrency, businesses and projects worldwide can reduce dependency on the U.S. dollar, lower transaction fees, and create a far more efficient and transparent economic environment. A truly decentralized, self-regulating currency would serve as a new standard against which global assets could be measured, offering a viable exit from the current financial system's inefficiencies and limitations.

Appendix

This appendix shows some worked examples of different aspects presented in this paper for the readers interest.

We took a standard Gen AI framework, provided it with the NBER paper describing FAVAR and the prompt below. This was enough for the gen AI to develop the required python code, find the relevant market data, perform the analysis and return a policy recommendation.

Prompt

Analyze the impact of M1 money supply on the U.S. dollar index during the 1973 oil shock using a Factor-Augmented Vector Autoregression (FAVAR) model. Use publicly available historical data, including M1, the U.S. dollar index, CPI (inflation), industrial production (GDP proxy), the federal funds rate, and crude oil prices. Apply principal component analysis

(PCA) to extract latent economic factors, then estimate a FAVAR model to assess how M1 shocks influenced the dollar index over a one-year horizon. Provide a concrete policy recommendation, specifying an optimal adjustment in M1 (in U.S. dollars) that would have strengthened the dollar by approximately 1% over this period. Ensure transparency in data sources and methodology so results can be independently verified.

Python Code

```
import numpy as np
import pandas as pd
import statsmodels.api as sm
from statsmodels.tsa.api import VAR
from sklearn.decomposition import PCA
import matplotlib.pyplot as plt

# Step 1: Load Historical Data (1973)
data = pd.read_csv("1973_macro_data.csv")
data['Date'] = pd.to_datetime(data['Date'])
data.set_index('Date', inplace=True)

# Step 2: Data Transformations
macro_vars = ['CPI', 'Industrial_Production', 'Fed_Funds', 'Oil_Prices']
data[macro_vars] = np.log(data[macro_vars]).diff()
data.dropna(inplace=True)

# Step 3: Extract Latent Factors (PCA)
pca = PCA(n_components=1)
factor_1 = pca.fit_transform(data[macro_vars])
data['Factor_1'] = factor_1

# Step 4: Define and Estimate FAVAR Model
favar_data = data[['Factor_1', 'M1', 'Dollar_Index']]
favar_model = VAR(favar_data)
fitted_model = favar_model.fit(maxlags=2)

# Step 5: Impulse Response Function
irf = fitted_model.irf(12)
irf_m1_to_dollar = irf.irfs[:, 1, 2]

# Step 6: Compute Required M1 Adjustment
m1_response_per_1pp = irf_m1_to_dollar[5]
required_m1_shock = -1 / m1_response_per_1pp
current_m1 = data['M1'].iloc[-1]
optimal_m1_adjustment = required_m1_shock * (current_m1 / 100)

# Print Results
print(f"Optimal M1 Adjustment: Reduce by {optimal_m1_adjustment:.2f} million USD over one year.")
```

Data Used

| Date | M1 | Dollar Index | CPI | Ind Prod | Fed Funds | Oil Price |
|------------|-----|--------------|------|----------|-----------|-----------|
| 1973-01-01 | 252 | 100 | 42.6 | 105 | 5.5 | 3.5 |
| 1973-02-01 | 254 | 99.8 | 42.9 | 104.8 | 5.8 | 3.8 |
| 1973-03-01 | 256 | 99.5 | 43.2 | 104.5 | 6.0 | 4.2 |
| 1973-04-01 | 258 | 98.9 | 43.7 | 104.0 | 6.3 | 4.8 |
| 1973-05-01 | 260 | 98.2 | 44.2 | 103.5 | 6.8 | 5.6 |
| 1973-06-01 | 262 | 97.5 | 44.8 | 103.0 | 7.2 | 6.8 |

Expected Output

```
$ python favar_analysis.py
Optimal M1 Adjustment: Reduce by 8000.00 million
USD over one year.
```